



**LUCAS VASCONCELOS SANTANA**

IME-USP

# APACHE STORM

is a free and open source distributed **realtime** computation system. Storm makes it easy to reliably process unbounded streams of data, doing for realtime processing what Hadoop did for batch processing. Storm is simple, can be used with any programming language, and is a lot of fun to use!

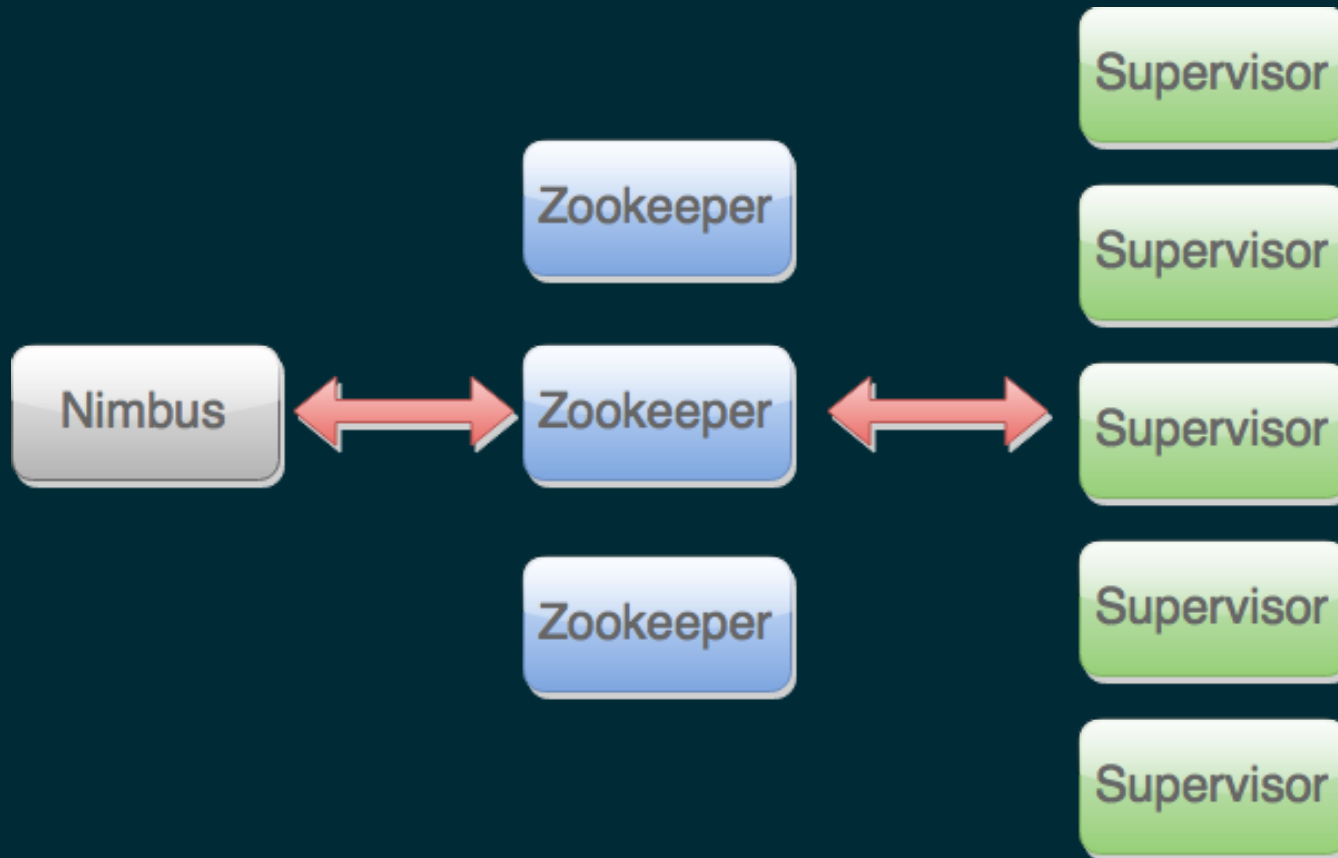
# INFOS

- Criado por Nathan Marz @ BackType
- Teve seu código aberto em 2011 após ser comprado pelo Twitter
- Em 2013 virou um projeto Apache (incubating)
- Apache Top-level project dia 29 de setembro de 2014
- ~15.000 linhas de código
- Maior parte escrita em Clojure
- 1M de mensagens/s (100 bytes cada) por nó

# COMPONENTES

- Zookeeper
- Nimbus
- Supervisor
- Topologias

# ARQUITETURA DE UM CLUSTER STORM



# TOLERÂNCIA A FALHAS

- "Stateless"
- Fail fast, auto restart
- Garante o processamento dos dados pelo menos uma vez

# TOPOLOGIA



Spouts

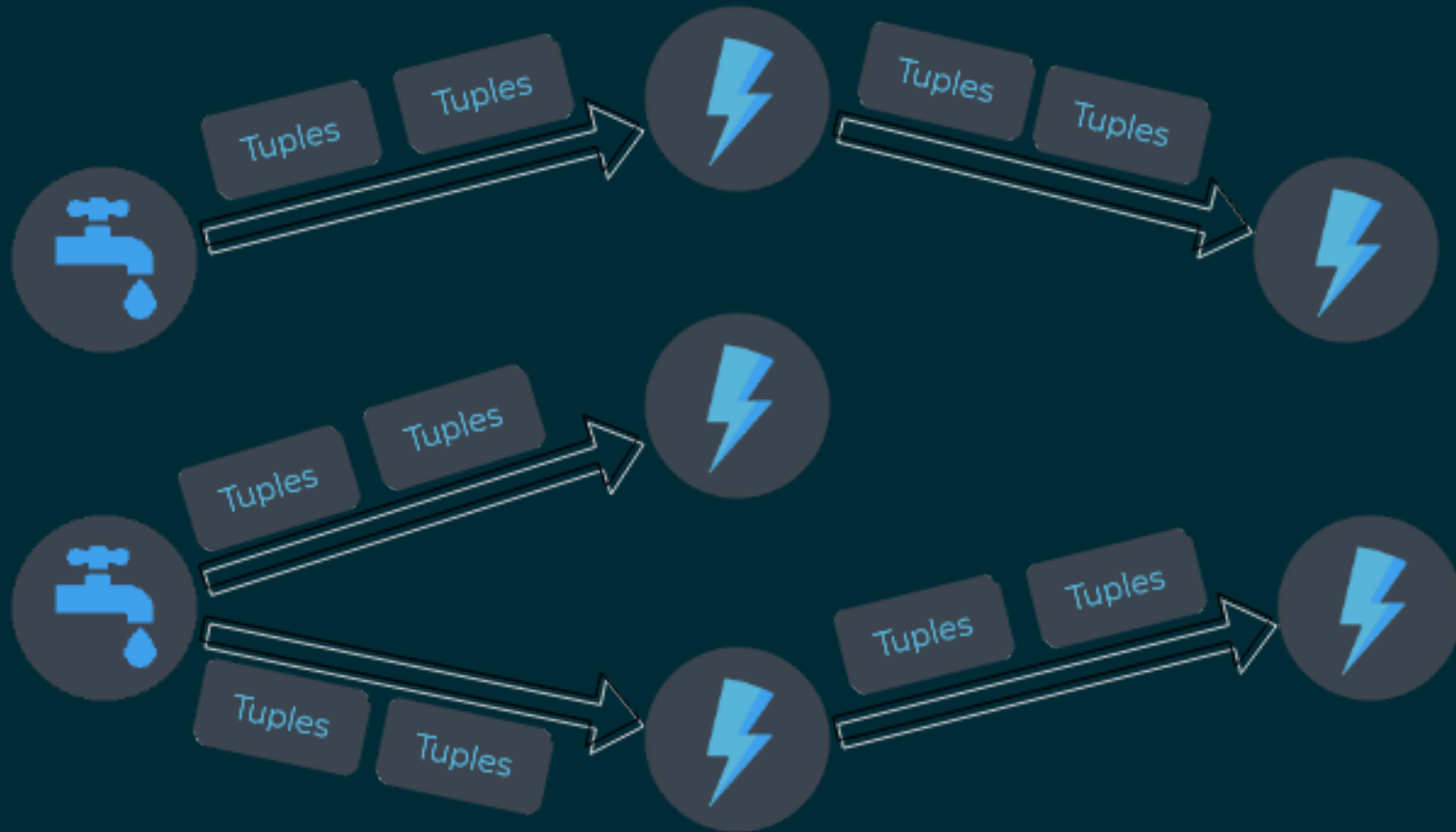


Bolts

Tuples

Tuples

# TOPOLOGIA





# SPOUTS

```
public static class TestWordSpout extends BaseRichSpout {  
  
    ...  
  
    public void nextTuple() {  
        Utils.sleep(100);  
        final String[] words = new String[] {"nathan", "mike", "jackson"};  
        final Random rand = new Random();  
        final String word = words[rand.nextInt(words.length)];  
        _collector.emit(new Values(word));  
    }  
}
```

# BOLTS

```
public static class ExclamationBolt implements IRichBolt {  
  
    ...  
  
    public void execute(Tuple tuple) {  
        _collector.emit(tuple, new Values(tuple.getString(0) + "!!!"));  
        _collector.ack(tuple);  
    }  
}
```

# EXECUTANDO A TOPOLOGIA

```
TopologyBuilder builder = new TopologyBuilder();
builder.setSpout("words", new TestWordSpout(), 10);
builder.setBolt("exclaim1", new ExclamationBolt(), 3)
    .shuffleGrouping("words");
builder.setBolt("exclaim2", new ExclamationBolt(), 2)
    .shuffleGrouping("exclaim1");
```

# GROUPINGS

- **Shuffle grouping:** distribuição aleatória das tuplas;
- **Field grouping:** mod hashing nas tuplas, enviando sempre para mesma task;
- **All grouping:** envia tupla para todas as tasks;
- etc.

# WORDCOUNT

```
TopologyBuilder builder = new TopologyBuilder();
builder.setSpout("sentences", new KestrelSpout("kestrel.backtype.com",
                                              22133,
                                              "sentence_queue",
                                              new StringScheme()));
builder.setBolt("split", new SplitSentence(), 10)
    .shuffleGrouping("sentences");
builder.setBolt("count", new WordCount(), 20)
    .fieldsGrouping("split", new Fields("word"));
```

# DEFININDO BOLTS EM OUTRAS LINGUAGENS

```
public static class SplitSentence extends ShellBolt implements IRichBolt
    public SplitSentence() {
        super("python", "splitsentence.py");
    }

    public void declareOutputFields(OutputFieldsDeclarer declarer) {
        declarer.declare(new Fields("word"));
    }
}
```

# DEFININDO BOLTS EM OUTRAS LINGUAGENS

```
import storm

class SplitSentenceBolt(storm.BasicBolt):
    def process(self, tup):
        words = tup.values[0].split(" ")
        for word in words:
            storm.emit([word])

SplitSentenceBolt().run()
```

# OUTROS USOS...

- Transaccional
- Distributed RPC



# REFERÊNCIAS

<http://storm.apache.org>

<http://www.infoq.com/presentations/Storm-Introduction>

<http://blog.spec-india.com/apache-storm-...-overall-compariso>

<http://nathanmarz.com/blog/history-...-lessons-learned.html>

[https://blogs.apache.org/.../the\\_apache\\_software\\_foundation\\_a](https://blogs.apache.org/.../the_apache_software_foundation_a)

A close-up photograph of Chuck Norris. He is wearing a dark-colored polo shirt with a name tag on the left side of his chest. He has a slight smile and is raising his right fist in a celebratory gesture. The background is dark and out of focus, with some wooden beams visible. The word "OBRIGADO!" is overlaid in white, bold, sans-serif font across the center of his face.

**OBRIGADO!**

**Chuck Norris**